UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/676,494	09/30/2003	Mark Moyer	M-15194 US	8571

7590 06/02/2006

Jon W. Hallman
MacPHERSON KWOK CHEN & HEID LLP
1762 Technology Drive, Suite 226
San Jose, CA 95110

EXAMINER	
CHAUDRY, MUJTABA M	

ART UNIT	PAPER NUMBER
2133	

DATE MAILED: 06/02/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/676,494	MOYER ET AL.	
	Examiner	Art Unit	
	Mujtaba K. Chaudry	2133	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *29 September 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-24* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *09 February 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *9/29/2005*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Information Disclosure Statement

The references listed in the information disclosure statement (IDS) submitted on September 29, 2205 and September 30, 2003 has been considered. The submission is in compliance with the provisions of 37 CFR 1.97.

Oath/Declaration

The Oath filed September 30, 2003 complies with all the requirements set forth in MPEP 602 and therefore is accepted.

Drawings

Figures 1-3 submitted February 9, 2004 are accepted.

Specification

The specification submitted September 30, 2003 is accepted.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claim 9 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the

invention. It is not clear how the checksum calculation engine and the checksum comparator are "implemented" by the logic core. It is understood the logic core to be a memory which stores configuration data. Clarification is requested.

Claim Rejections - 35 USC § 103

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

Claims 1-24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Plants (USPN 6237124) further in view of Carmichael et al. (USPN 7036059).

As per claim 1, Plants substantially teaches (Figure 4 and col. 5) a FPGA 28 (analogous to programmable logic device) comprising a FPGA core 10 (analogous to configuration memory storing configuration data). Plants teaches (col. 4 and abstract) to perform cyclic redundancy check using cyclic redundancy circuit 40 to verify the integrity of the configuration data in the FPGA core 10. See Figure 4:

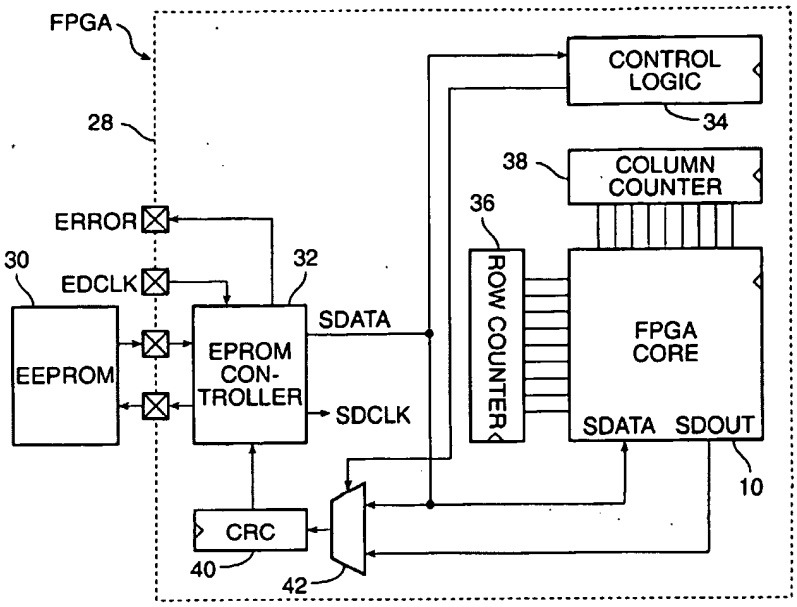

FIG. 4

Plants does not explicitly teach to calculate a checksum for the configuration data in the FPGA core as stated in the present application.

However, Carmichael et al. (herein after: Carmichael), in an analogous art, substantially teaches (col., 23, lines 20-30) to compare checksum in configuration data for a FPGA. Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to calculate a checksum for the configuration data in the FPGA in place of CRC as is taught by Plants. This modification would have been obvious to one of ordinary skill in the art at the time the invention was made because one of ordinary skill in the art would have recognized that the calculation process for the checksum and CRC are compatible, since a checksum is obtained by adding the digits in a numeral without regard to position or meaning. This sum is then compared to a predetermined checksum. In contrast, CRC is encoded within the data and is based on the

data itself. Essentially, both are well known mechanisms for detecting faults in configuration data stored in FPGAs.

As per claim 2, Plants substantially teaches, in view of above rejections, the checksum calculation to be performed by the CRC 40 shown in Figure 4.

As per claim 3, Carmichael substantially teaches (col. 2, lines 1-15), in view of above rejections, to use parity calculation in determining the errors in the configuration data of the FPGA.

As per claims 4 and 8, Carmichael substantially teaches (Figure 13 and col. 23, lines 20-35), in view of above rejection, to store a first type of configuration data and a second type of configuration data. Carmichael teaches a way of using the dual device dual logic technique in an alternative architecture. A dual voting system 84, based on duplicate logic functions, incorporates into two FPGAs 86, 87 and a storage PROM 88 a fully redundant, self-mitigating system with built-in SEU detection and correction. The system 84 further comprises the user's basic design (logic) 90, 91; duplicates of the basic design (duplicate logic) 92, 93; a STARTUP component (primitive) 94, 95; a constant Low output 96, 97; a falling edge detector 98, 99, support logic 100, 101; and a state machine 106, 107 to control readback of configuration memory and auto-configuration of the neighboring FPGA 86, 87.

As per claim 5, Plants substantially teaches, in view of above rejections, the checksum calculation to be performed by the CRC 40 shown in Figure 4, which the Examiner would like to point out is well known to comprise LFSRs.

As per claims 6-7, Plants substantially teaches, in view of above rejections, the checksum calculation to be performed by the CRC 40 shown in Figure 4. Plants teaches to compare a

predetermined CRC with the calculated CRC to verify the FPGA. Therefore the CRC 40 unit inherently has to have a register for storing the predetermined CRC in order to perform comparison analysis.

As per claim 9, Carmichael substantially teaches (Figure 13 and col. 23, lines 20-35), in view of above rejection, to store a first type of configuration data and a second type of configuration data. Carmichael teaches a way of using the dual device dual logic technique in an alternative architecture. A dual voting system 84, based on duplicate logic functions, incorporates into two FPGAs 86, 87 and a storage PROM 88 a fully redundant, self-mitigating system with built-in SEU detection and correction. The system 84 further comprises the user's basic design (logic) 90, 91; duplicates of the basic design (duplicate logic) 92, 93; a STARTUP component (primitive) 94, 95; a constant Low output 96, 97; a falling edge detector 98, 99, support logic 100, 101; and a state machine 106, 107 to control readback of configuration memory and auto-configuration of the neighboring FPGA 86, 87. See rejection under 35 USC 112.

As per claim 10, Plants substantially teaches, in view of above rejections, the checksum calculation to be performed by the CRC 40 shown in Figure 4. Plants teaches to compare a predetermined CRC with the calculated CRC to verify the FPGA. Therefore the CRC 40 unit inherently has to have a register for storing the predetermined CRC in order to perform comparison analysis. The Examiner would like to point out that CRC 40, which performs CRC calculation and comparison is on dedicated circuitry within the FPGA.

As per claim 11, Plants substantially teaches (Figure 4 and col. 5) a FPGA 28 (analogous to programmable logic device) comprising a FPGA core 10 (analogous to configuration memory

storing configuration data). Plants teaches (col. 4 and abstract) to perform cyclic redundancy check using cyclic redundancy circuit 40 to verify the integrity of the configuration data in the FPGA core 10. See Figure 4.

Plants does not explicitly teach to calculate a checksum for the configuration data in the FPGA core as stated in the present application.

However, Carmichael et al. (herein after: Carmichael), in an analogous art, substantially teaches (col., 23, lines 20-30) to compare checksum in configuration data for a FPGA. Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to calculate a checksum for the configuration data in the FPGA in place of CRC as is taught by Plants. This modification would have been obvious to one of ordinary skill in the art at the time the invention was made because one of ordinary skill in the art would have recognized that the calculation process for the checksum and CRC are compatible, since a checksum is obtained by adding the digits in a numeral without regard to position or meaning. This sum is then compared to a predetermined checksum. In contrast, CRC is encoded within the data and is based on the data itself. Essentially, both are well known mechanisms for detecting faults in configuration data stored in FPGAs.

As per claim 12, Plants substantially teaches, in view of above rejections, the checksum calculation to be performed by the CRC 40 shown in Figure 4.

As per claim 13, Plants substantially teaches, in view of above rejections, the checksum calculation to be performed by the CRC 40 shown in Figure 4. Plants teaches to compare a predetermined CRC with the calculated CRC to verify the FPGA. Therefore the CRC 40 unit

inherently has to have a register for storing the predetermined CRC in order to perform comparison analysis.

As per claim 14, Carmichael substantially teaches (Figure 13 and col. 23, lines 20-35), in view of above rejection, to store a first type of configuration data and a second type of configuration data. Carmichael teaches a way of using the dual device dual logic technique in an alternative architecture. A dual voting system 84, based on duplicate logic functions, incorporates into two FPGAs 86, 87 and a storage PROM 88 a fully redundant, self-mitigating system with built-in SEU detection and correction. The system 84 further comprises the user's basic design (logic) 90, 91; duplicates of the basic design (duplicate logic) 92, 93; a STARTUP component (primitive) 94, 95; a constant Low output 96, 97; a falling edge detector 98, 99, support logic 100, 101; and a state machine 106, 107 to control readback of configuration memory and auto-configuration of the neighboring FPGA 86, 87.

As per claim 15, Plants substantially teaches (Figure 4 and col. 5) a FPGA 28 (analogous to programmable logic device) comprising a FPGA core 10 (analogous to configuration memory storing configuration data). Plants teaches (col. 4 and abstract) to perform cyclic redundancy check using cyclic redundancy circuit 40 to verify the integrity of the configuration data in the FPGA core 10. See Figure 4.

Plants does not explicitly teach to calculate a checksum for the configuration data in the FPGA core as stated in the present application.

However, Carmichael et al. (herein after: Carmichael), in an analogous art, substantially teaches (col., 23, lines 20-30) to compare checksum in configuration data for a FPGA. Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made

to calculate a checksum for the configuration data in the FPGA in place of CRC as is taught by Plants. This modification would have been obvious to one of ordinary skill in the art at the time the invention was made because one of ordinary skill in the art would have recognized that the calculation process for the checksum and CRC are compatible, since a checksum is obtained by adding the digits in a numeral without regard to position or meaning. This sum is then compared to a predetermined checksum. In contrast, CRC is encoded within the data and is based on the data itself. Essentially, both are well known mechanisms for detecting faults in configuration data stored in FPGAs.

As per claims 16-17, Plants substantially teaches, in view of above rejections, the checksum calculation to be performed by the CRC 40 shown in Figure 4. Plants teaches to compare a predetermined CRC with the calculated CRC to verify the FPGA. Therefore the CRC 40 unit stores the predetermined CRC in order to perform comparison analysis within the FPGA as shown in Figure 4.

As per claim 18, Carmichael substantially teaches (Figure 13 and col. 23, lines 20-35), in view of above rejection, to store a first type of configuration data and a second type of configuration data. Carmichael teaches a way of using the dual device dual logic technique in an alternative architecture. A dual voting system 84, based on duplicate logic functions, incorporates into two FPGAs 86, 87 and a storage PROM 88 a fully redundant, self-mitigating system with built-in SEU detection and correction. The system 84 further comprises the user's basic design (logic) 90, 91; duplicates of the basic design (duplicate logic) 92, 93; a STARTUP component (primitive) 94, 95; a constant Low output 96, 97; a falling edge detector 98, 99,

support logic 100, 101; and a state machine 106, 107 to control readback of configuration memory and auto-configuration of the neighboring FPGA 86, 87.

As per claims 19-20, Plants substantially teaches, in view of above rejections, the checksum calculation to be performed by the CRC 40 shown in Figure 4. Plants teaches to compare a predetermined CRC with the calculated CRC to verify the FPGA. This method verifies the integrity of the FPGA's configuration data and accordingly flags the corrupted data.

As per claims 21-23, Plants substantially teaches (Figure 4 and col. 5) a FPGA 28 (analogous to programmable logic device) comprising a FPGA core 10 (analogous to configuration memory storing configuration data). Plants teaches (col. 4 and abstract) to perform cyclic redundancy check using cyclic redundancy circuit 40 to verify the integrity of the configuration data in the FPGA core 10. See Figure 4.

Plants does not explicitly teach to calculate a checksum for the configuration data in the FPGA core as stated in the present application.

However, Carmichael et al. (herein after: Carmichael), in an analogous art, substantially teaches (col., 23, lines 20-30) to compare checksum in configuration data for a FPGA. Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to calculate a checksum for the configuration data in the FPGA in place of CRC as is taught by Plants. This modification would have been obvious to one of ordinary skill in the art at the time the invention was made because one of ordinary skill in the art would have recognized that the calculation process for the checksum and CRC are compatible, since a checksum is obtained by adding the digits in a numeral without regard to position or meaning. This sum is then compared to a predetermined checksum. In contrast, CRC is encoded within the data and is based on the

data itself. Essentially, both are well known mechanisms for detecting faults in configuration data stored in FPGAs.

As per claim 24, Plants substantially teaches (Figure 4 and col. 5) a FPGA 28 (analogous to programmable logic device) comprising a FPGA core 10 (analogous to configuration memory storing configuration data). Plants teaches (col. 4 and abstract) to perform cyclic redundancy check using cyclic redundancy circuit 40 to verify the integrity of the configuration data in the FPGA core 10. See Figure 4.

Plants does not explicitly teach to calculate a checksum for the configuration data in the FPGA core as stated in the present application.

However, Carmichael et al. (herein after: Carmichael), in an analogous art, substantially teaches.(col., 23, lines 20-30) to compare checksum in configuration data for a FPGA. Therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made to calculate a checksum for the configuration data in the FPGA in place of CRC as is taught by Plants. This modification would have been obvious to one of ordinary skill in the art at the time the invention was made because one of ordinary skill in the art would have recognized that the calculation process for the checksum and CRC are compatible, since a checksum is obtained by adding the digits in a numeral without regard to position or meaning. This sum is then compared to a predetermined checksum. In contrast, CRC is encoded within the data and is based on the data itself. Essentially, both are well known mechanisms for detecting faults in configuration data stored in FPGAs.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Additional pertinent prior arts are included herein for Applicant's review.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Mujtaba K. Chaudry whose telephone number is 571-272-3817. The examiner can normally be reached on Mon-Thur 9-7:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Albert DeCady can be reached on 571-272-3819. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Mujtaba Chaudry
Art Unit 2133
May 26, 2006

GUY LAMARRE
PRIMARY EXAMINER